



# 实验3 多功能ALU设计实验

1



- 一、实验目的
- 二、实验原理与实验内容
- 三、实验要求
- 四、实验步骤
- 五、思考与探索





# 一、实验目的



- 学习**多功能ALU**的工作原理，掌握**多功能ALU**的设计方法；
- 掌握**暂存器**的设计方法，及其**与ALU**的连接方法；
- 掌握运用Verilog HDL语言进行组合逻辑电路与时序逻辑电路混合设计的方法；
- 学会**输入输出设备不足**时的处理方法。





## 二、实验内容与原理



### ■ 实验内容：

- 设计一个能够满足RV32I指令集需求的、具有10种运算功能的32位ALU，并能够产生运算结果的标志ZF、CF、OF和SF；
- 设计暂寄存器A、B、F及标志寄存器FR，并且和ALU进行连接。

1、多功能ALU

2、运算结果的标志位

3、暂寄存器和标志寄存器FR

4、构造用于板级验证的顶层测试模块



# 1、多功能ALU

## ■ 功能:

■ 功能定义: 取决于目标指令系统 (RV32I)

■ 运算功能: 10种, 由 **ALU\_OP[3:0]** 来选择

■ 运算宽度: 32位

■ **slt**: set if less than

■ **sltu**: set if less than for unsigned value

■ **sra**: shift right arithmetically

ALU_OP [3:0]	运算	助记符
0000	加法: $A+B$	add
0001	左移: $A \ll B$	sll
0010	有符号数比较 ( $A < B$ ) ? 1 : 0	slt
0011	无符号数比较 ( $A < B$ ) ? 1 : 0	sltu
0100	异或: $A \wedge B$	xor
0101	逻辑右移: $A \gg B$ , 高位补0	srl
0110	按位或: $A   B$	or
0111	按位与: $A \& B$	and
1000	减法 $A-B$	sub
1101	算术右移: $A \ggg B$ , 高位补 $A[31]$	sra

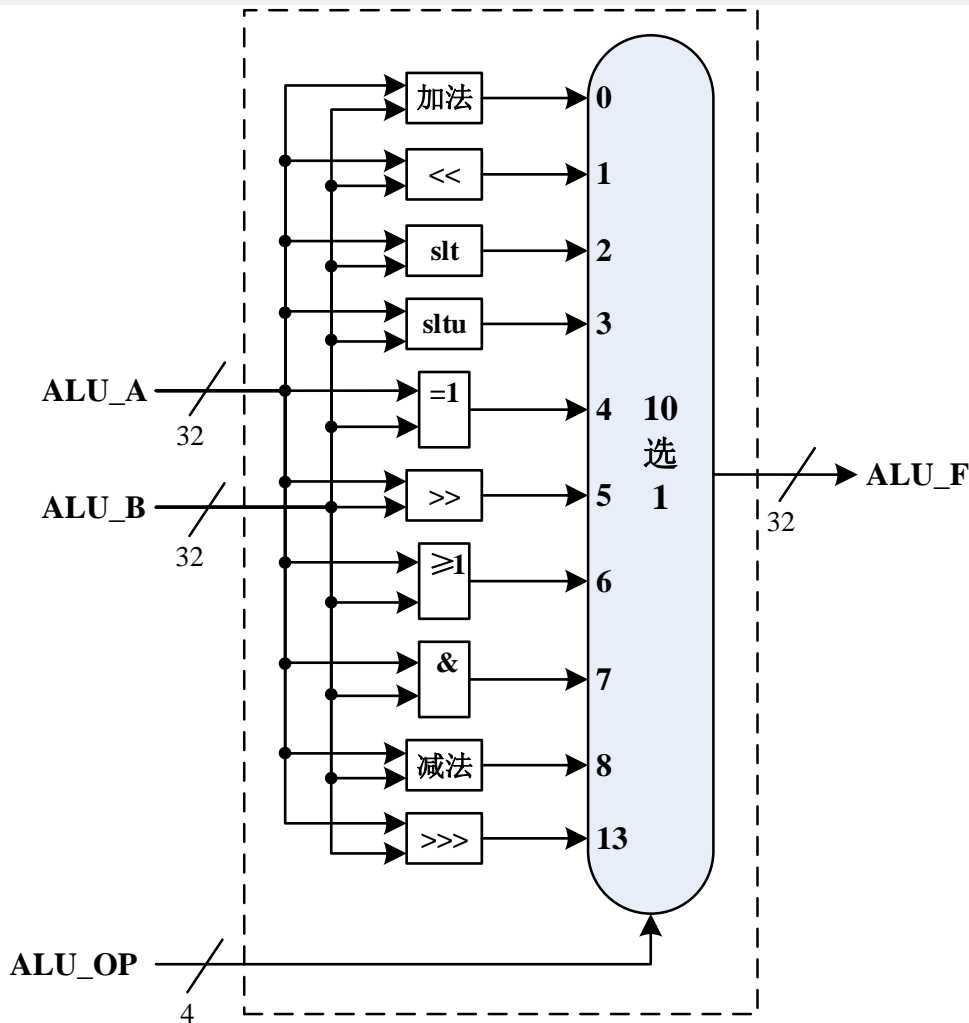
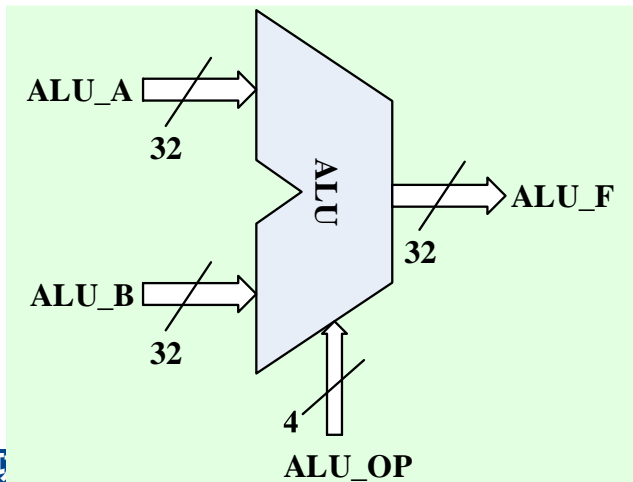


# 1、多功能ALU

## ■ 硬件结构

■ 运算部件：10种运算功能由多个部件并行实现

■ 十选一数据选择器：选通输出运算结果；选择线为ALU\_OP[3:0]





# 1、多功能ALU

## ■ Verilog HDL实现:

### ■ ①直接使用运算操作符

■ 加法:  $\{C32,F\}=A+B;$

■ 减法:  $\{C32,F\}=A-B;$

### ■ ②区分有符号和无符号数?

■ 默认: 数据是无符号数

■ 有符号数: 用\$signed来指定

**\$signed(Data):** 指定Data是有符号数 (补码表示)

ALU_OP [3:0]	运算	助记符
0000	加法: $A+B$	and
0001	左移: $A<<B$	sll
0010	有符号数比较 $(A<B) ? 1:0$	slt
0011	无符号数比较 $(A<B) ? 1:0$	sltu
0100	异或: $A^B$	xor
0101	逻辑右移: $A>>B$ , 高位补0	srl
0110	按位或: $A B$	or
0111	按位与: $A\&B$	and
1000	减法 $A-B$	sub
1101	算术右移: $A>>>B$ ,高位补 $A[31]$	sra



## 2、运算结果的标志位

7



C32?

### ①结果为零标志ZF (Zero Flag)

- 结果为全零:  $ZF=1$
- 结果不全为零:  $ZF=0$

$$ZF = \overline{F_{31} + F_{30} + \cdots + F_1 + F_0}$$

### ②符号标志SF (Sign Flag)

- 结果为正数时, SF置为0, 否则其值为1

$$SF = F_{31}$$

### ③进位/借位标志CF (Carry Flag)

- 加法运算有进位, 减法运算有借位:  $CF=1$ ; 否则 $CF=0$
- 加法运算:  $C=1$ 则 $CF=1$ , 否则 $=0$
- 减法运算:  $C=0$ 则 $CF=1$ , 否则 $=0$
- C: 硬件运算时最高位产生的进位
- CF只对无符号数有意义 (描述无符号数加法运算是否溢出, 减法运算是否不够减)

$$CF = \overline{ADD/SUB} \cdot C + \overline{ADD/SUB} \cdot \bar{C}$$



## 2、运算结果的标志位

8



### ④溢出标志OF (Overflow Flag)

- 有符号数运算结果超出表示范围:  
OF=1, 否则OF=0
- OF只对有符号数有意义

当有符号数运算发生溢出时  
(OF=1), SF是错误的

$$OF = C_{32} \oplus C_{31} \quad (\text{进位判溢})$$

双符号位判溢?

$$\begin{array}{r} A_{31} A_{31} A_{30} \dots \dots A_1 A_0 \\ + B_{31} B_{31} B_{30} \dots \dots B_1 B_0 \\ \hline F_{32} F_{31} F_{30} \dots \dots F_1 F_0 \end{array}$$

$$\begin{aligned} OF &= F_{32} \oplus F_{31} \\ &= A_{31} \oplus B_{31} \oplus C_{32} \oplus F_{31} \end{aligned}$$



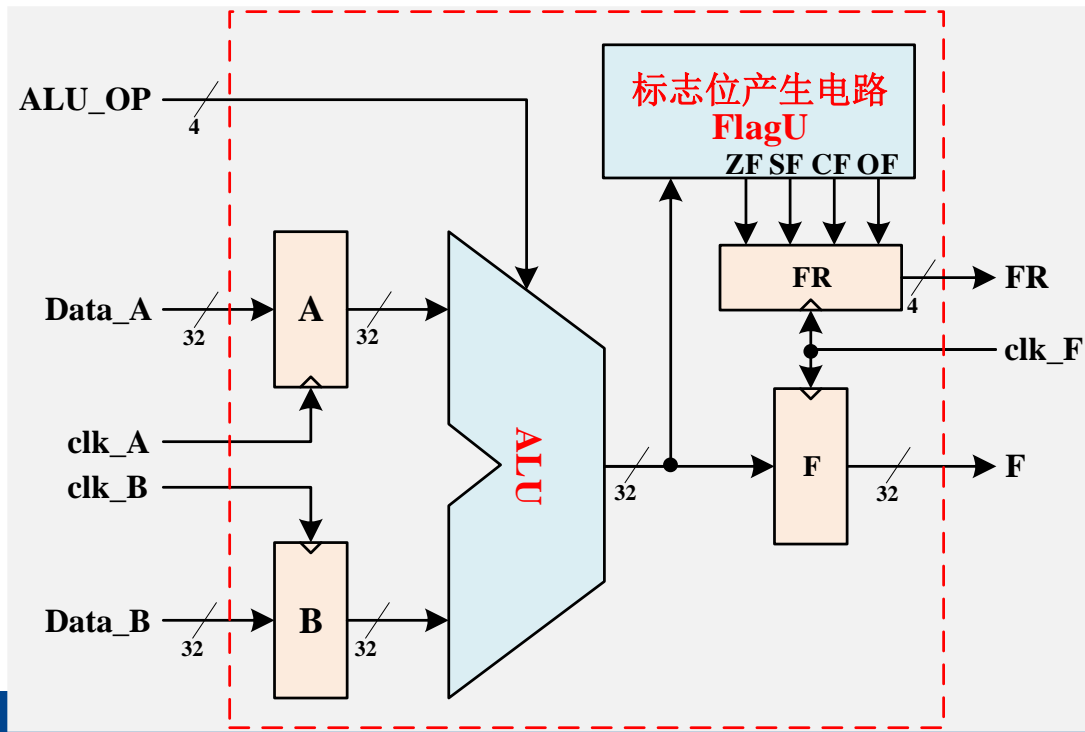


### 3、暂存器和标志寄存器FR

9



- 暂存器：将多功能ALU的输入数据和输出数据暂时存储到**锁存器或者寄存器中**
- **A、B**：保存ALU输入数据
- **F**：保存ALU运算结果
- **FR**：保存4个标志位
- 三个时钟信号





- 

## //打入数据



## 4、构造用于板级验证的顶层测试模块

11



### ■ 板级验证：

- **目的：**构造用于板级验证的顶层测试模块，来验证设计是否正确
- **设计实验方案：**模块的输入信号约束到输入设备（开关、按键），输出信号约束到输出设备（LED显示灯和数码管）
  - 输入的数据信号：是电平信号，通常配置到逻辑开关；
  - 输入的时钟和复位信号：是边沿信号，通常配置到按键；
  - 输出的数据信号：既可以配置到LED灯以二进制显示，也可以送到数码管以十六进制形式或者转换成十进制显示；
  - 输出的状态信号：通常送到LED灯显示，这样可读性较好。






## 4、构造用于板级验证的顶层测试模块

12



- **实验板卡：**输入输出设备数量不同，但都是有限的
- **问题：**输入输出设备数量不足，不能一次性输入所有的输入数据或者一次性展示所有的执行结果
- **解决方案：**分时复用I/O设备

■ 常见的板卡型号：

设备	HDU-XL-01	HCS-A01	Nexys4
开关	32个	20个	16个
按键	5个 	8个 	5个 
显示灯	32个	20个	16个
数码管	8个	8个	8个

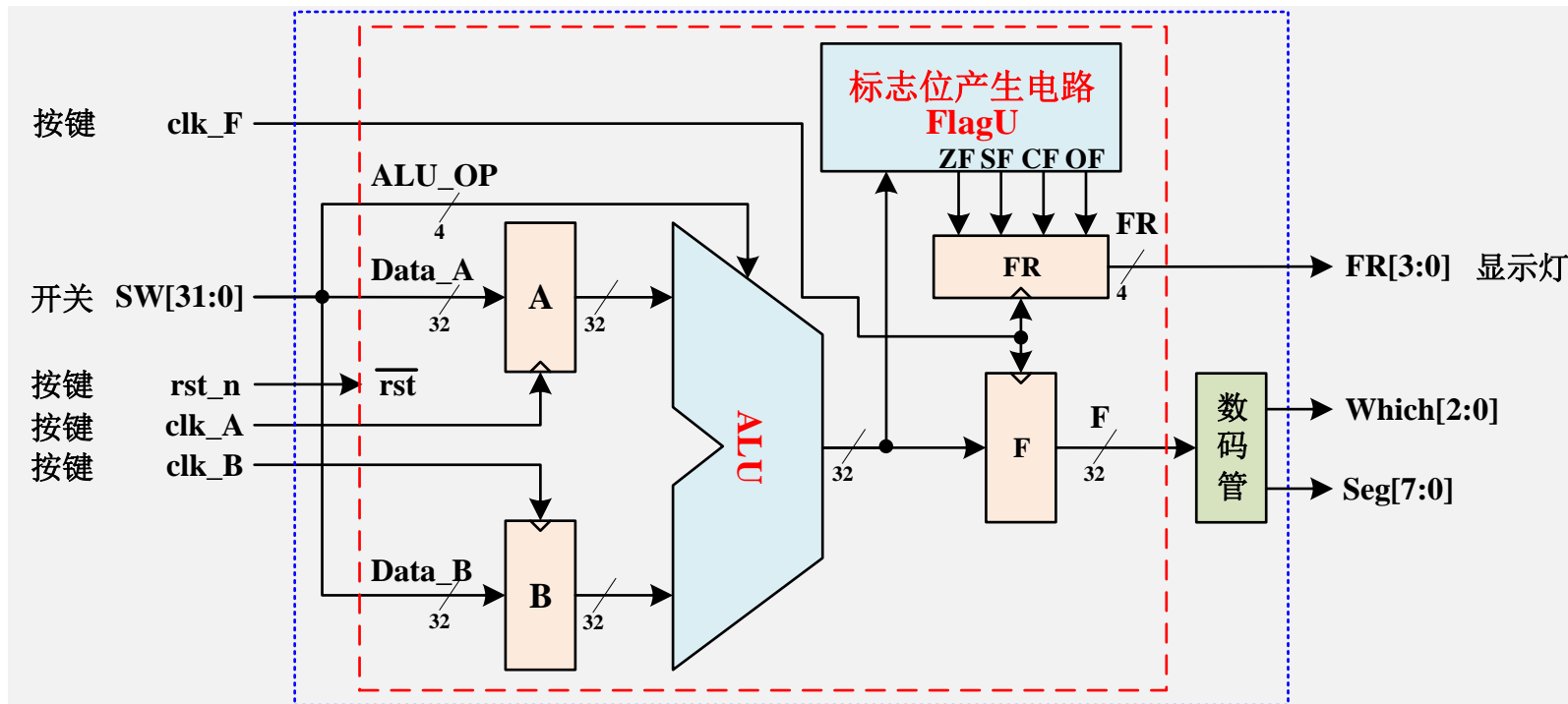


# ① HDU-XL-01板级验证

13



- 有： 开关32个、按键5个、LED灯32个、数码管8个
- 需要： 开关32个、按键4个、LED灯4个、数码管8个

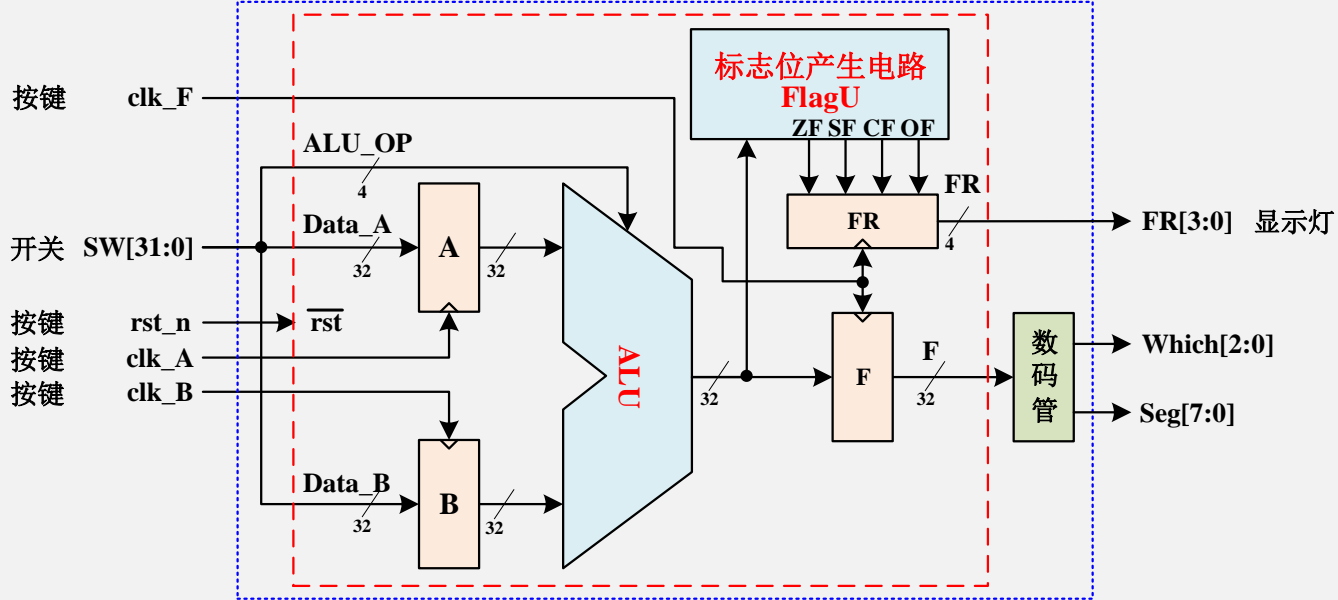




# ① HDU-

■ 32位开关:  
Data\_A、  
Data\_B、  
ALU\_OP分时复用

## ■ 操作过程



①按rst\_n按键: 0→A、B、F、FR

②拨32位开关为数据Data\_A, 按clk\_A按键, 数据打入A暂存器; Data\_A→A

③拨32位开关为数据Data\_B, 按clk\_B按键, 数据打入B暂存器; Data\_B→B

④拨ALU\_OP对应的4位开关, 选择运算功能 $\theta$ , 然后按clk\_F按键, ALU运算结果  
(=Data\_A  $\theta$  Data\_B) 打入F暂存器; Data\_A  $\theta$  Data\_B→F

⑤观察数码管显示的运算结果, 以及显示灯显示的标志位, 验证是否正确。

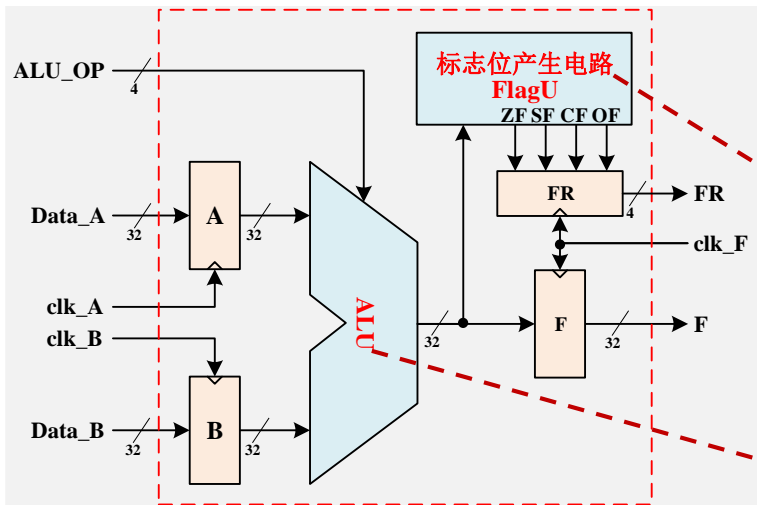


# 三、实验要求

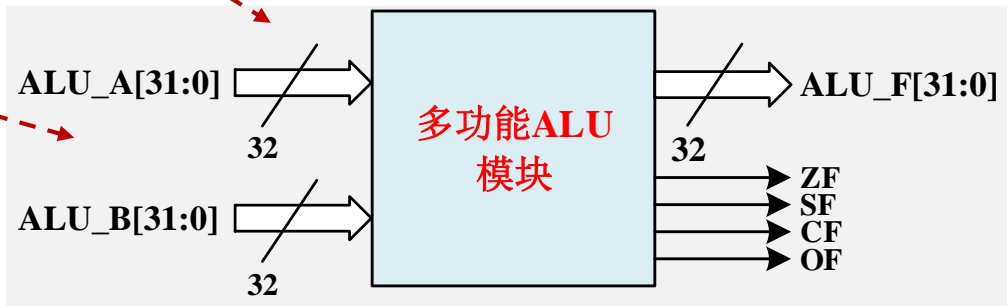
15



1. 使用行为描述方式建模**多功能ALU模块**，并添加标志位产生电路**FlagU**。仿真验证其功能。



模块引脚图



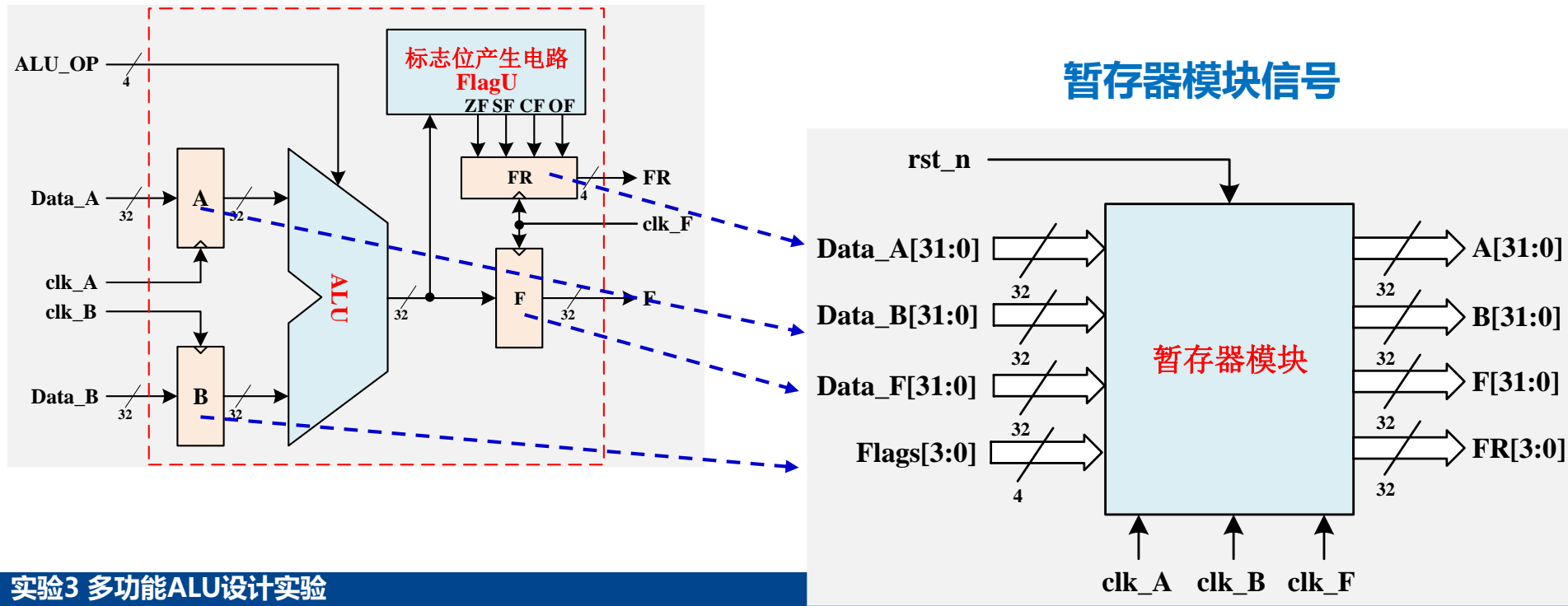


## 三、实验要求

16



2. 设计一个**暂存器模块**，包含A、B、F和标志寄存器FR的寄存电路及控制逻辑（这里也可设计为各自独立的4个模块）。





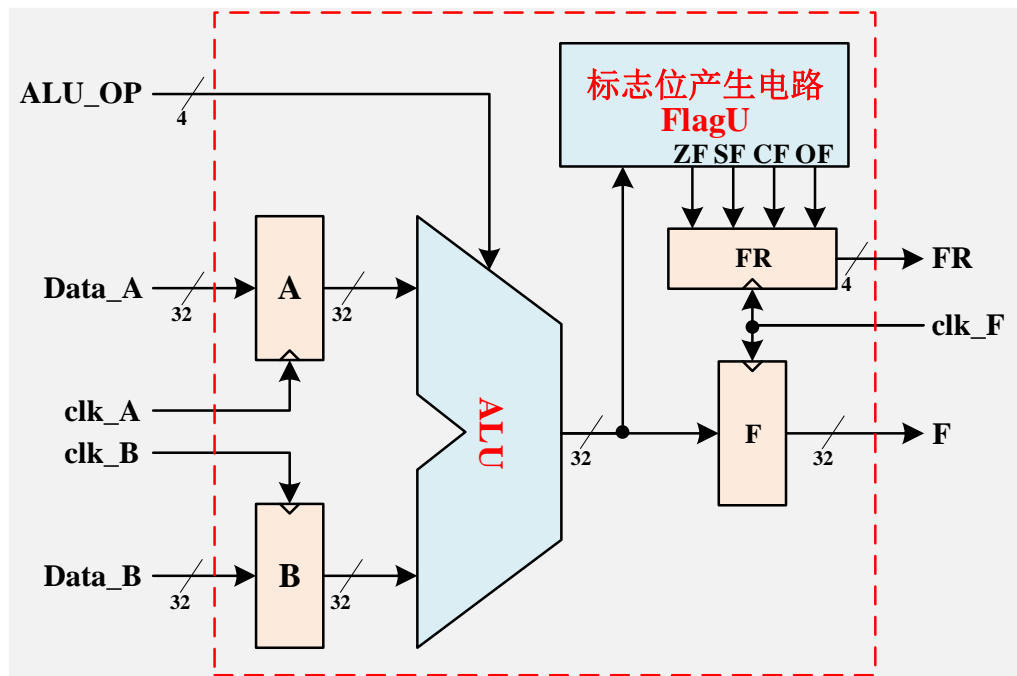


### 三、实验要求

17



3. 编写一个带暂存器的多功能ALU模块，将多功能ALU模块与暂存器模块连接在一起。
4. 针对使用的实验板卡，设计带暂存器的多功能ALU模块的板级验证实验方案，编写顶层测试模块；





## 三、实验要求



5. 选择典型的输入数据，验证你的ALU的各种功能，将实验结果记录到表格，要求所有运算功能和标志位都被有效测试。

Data_A[31:0]	Data_B[31:0]	ALU_OP	功能	F[31:0]	FR (ZF SF CF OF)



## 三、实验要求

19



6. 撰写实验报告（不做要求），格式见附录，重点内容包括：

- 对仿真结果进行分析；
- 描述你设计的板级验证实验方案、模块结构与连接；
- 说明你的板级操作过程；
- 分析记录下来的板级实验结果、得到有效结论。
- 请力所能及回答或实践本实验的“思考与探索”部分。





## 四、实验步骤

20



1. 新建工程，编写多功能**ALU模块**的代码，包括**标志位的产生电路**。
2. 编写激励代码，**仿真测试多功能ALU模块**；分析仿真结果，确保模块运算逻辑正确、标志位产生逻辑正确。
3. 新建一个**暂存器模块**，包含A、B、F三个32位暂存器和一个4位的标志寄存器FR。
4. 新建一个**带暂存器的多功能ALU模块**，将上述两个模块连接起来。
5. 设计板级验证的实验方案，然后据此编写一个**顶层测试模块**。
6. 可以依据实际需要，对顶层测试模块进行仿真测试，或者直接进入管脚约束环节。



## 四、实验步骤



7. 新建**管脚约束**文件，依据设计的板级验证实验方案，进行相应的引脚配置。
8. 生成\*.bit文件，下载到实验设备的FPGA芯片中。
9. **板级实验**：按照你所设计的实验方案，操作输入设备、观察输出设备，一般过程为：
  - 1) 按rst\_n键复位；
  - 2) 拨动开关输入数据A，按下时钟键clk\_A，打入暂存器A；
  - 3) 拨动开关输入数据B，按下时钟键clk\_B，打入暂存器B；
  - 4) 拨动开关选择运算功能ALU\_OP，按下时钟键clk\_F，保存结果到F；
  - 5) 拨动开关选择输出的数据，观察LED灯或者数码管，记录实验结果到表中，并分析实验结果是否正确。





## 五、思考与探索（至少完成1道）

22



1. 你是如何实现有符号数的比较置位运算slt和无符号数的比较置位运算sltu？你使用什么数据来验证你的slt和sltu运算是正确的？
2. 你是如何实现逻辑右移srl运算和算术右移sra运算的？同样，你使用什么数据来验证你的srl和sra运算是正确的？
3. CF的硬件逻辑表达式为 $CF = \overline{ADD/SUB} \cdot C + \overline{ADD/SUB} \cdot \bar{C}$ ，如果采用Verilog HDL语言的运算符进行加法和减法运算，如下所示：  
 $\{C32, ALU\_F\} = ALU\_A - ALU\_B;$   
 $\{C32, ALU\_F\} = ALU\_A + ALU\_B;$   
这里的C32是式中的C吗？请通过仿真验证当 $ALU\_A = 32'h7FFF\_FFF0$ ， $ALU\_B = 32'h7FFF\_FFFF$ 时，加减法运算后，CF的逻辑是否正确，从而给出答案。



## 五、思考与探索 (至少完成1道)

23



4. 你的板级验证实验，使用了什么板卡？输入输出设备够吗？如果不够，你是如何解决的？你用了什么输出设备显示运算结果？能观察什么数据？说说你的方案。
5. RV32I指令集中有一条指令blt，功能是：有符号数比较，小于则跳转（Branch if Less Than）。实现的方法是：先做减法，再根据运算结果的标志位进行判断是否小于，从而决定是否转移（分支）。请分析，**blt指令根据什么标志位、如何来判定。**
6. 如果是无符号数比较小于跳转指令**bltu**呢？
7. 如果你使用了图示的**A和B指定8组常数**这种方案，请分析**可能的运算组合有多少？为什么？**
8. 谈谈你在实验中碰到了哪些问题？又是如何解决的？

